REMARKS/ARGUMENTS

Claim 1 is amended to fix a typographical error. No claims are canceled or added. Thus, claims 1-21 remain pending.

Claim Rejections under 35 USC § 103(a), Murai, Frederickson, Tamai, Hashemi

Claims 1-11, 13-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Murai (U.S. Pat. No. 4,866,717) in view of Frederickson (U.S. Pat. No. 5,805,799) in view of Tamai (U.S. Pat. No. 6,799,283) and further in view of Hashemi (U. S. Pat. No. 6,981,171).

Claim 1 is allowable over the cited references, either alone or in combination, as those references fail to teach or suggest all the elements of claim 1. For example, claim 1 recites:

> *a cycle redundancy check (CRC) engine that generates old cyclic redundancy check bytes based on the old LBA , performs a first exclusive OR (XOR) function only on the old LBA and the new LBA, performs Galois Field multiplication on a result of the first XOR function, and performs a second XOR function on a result of the Galois Field multiplication and the old cyclic redundancy check bytes to generate updated cyclic redundancy check bytes that are based on the new LBA.*

Murai and Tamai

At page 2, the Office Action asserts that the combination of Murai and Tamai discloses a controller that reassigns a sector of data from an old logical block address (LBA) to a new LBA when the portion of the disk corresponding to the old LBA contains a defect. However, the Office Action notes that Murai and Tamai do not disclose a CRC engine, as recited in claim 1. Frederickson and Hashemi are asserted to teach the recited CRC engine.

Frederickson

At page 3, the Office Action states:

> However, in the same field of endeavor, Frederickson (fig. 4) discloses data integrity code including logical block address including a CRC that performs Galois Field multiplication on a result of the first XOR function, and performs XOR function on a result of the Galois Field multiplication (col. 2, lines 37-47; col. 10, lines 60 through col. 11 line 38).

This statement misses a fundamental difference between the circuit 200 in FIG. 4 of Frederickson and CRC engine of claim 1. Although circuit 200 does indeed have a first XOR, a Galois Field multiplier, and a second XOR, the inputs of circuit 200 are different from the

inputs recited in claim 1, and other structural differences exist as well. Exemplary differences are detailed in pages 6-8 of the response filed August 3, 2007.

For example, Fredrickson is directed to a method of appending LBA information to user data *u* to create a data integrity block (DIB) word x_N to enable data block LBA verification during a block transfer process. *See Fredrickson*, abstract and col. 5 lines 7-22. There is no mention of using old CRC or other check bytes to generate updated CRC bytes, as recited in claim 1.

Furthermore, XOR gate array 203 operates on user data and not on LBA data as is asserted. In contrast, claim 1 recites performing "*a first exclusive OR (XOR) function only on the old LBA and the new LBA*." Additionally, XOR 232, which is the asserted second XOR function, receives the output of the Galois multiplier 225 and the LBA data 207. *Id.*, FIG. 4 and col. 11 lines 8-14. Thus, Frederickson only uses the current LBA word, including during the first cross-check during a read operation, and an "*old LBA*" is never used by either XOR. *Id.*, col. 7 line 56 to col. 8 line 4.

Moreover, XOR 232 receives the output of the multiplier 225 and the LBA data on line 207, and not an old check byte. Accordingly, Frederickson does not teach or suggest performing "*a second XOR function on a result of the Galois Field multiplication and the old cyclic redundancy check bytes*," as recited in claim 1.

The fact that one circuit contains a few of the same basic logic units as another circuit by itself does not teach or suggest that the two circuits perform the same functionality, or an obvious variant thereof. As circuit 200 provides different functional results than those recited in claim 1, additional supporting information must be provided as to why and how the inputs and other structural differences would be modified to provide the CRC engine of claim 1.

Hashemi

At page 3, the Office Action asserts that Hashemi teaches such a modification to the circuit 200 of Frederickson. However, Hashemi is not directed to calculating a CRC, and only provides a very minimal discussion about generating a CRC, which does not include using old CRC bytes to generate new CRC bytes, particularly those based on a new LBA..

Hashemi is directed to a more efficient initialization of parity data along a stripe of a RAID, where each block of a stripe is on a different hard drive. *See Hashemi*, col. 3 lines 6-18. Parity data is used to reconstruct data on another drive when that another drive goes bad. *Id.*, col. 5 lines 28-31. Parity data may be created by a read-modify-write" scheme, where when new data is written to a stripe, the old parity data for that stripe is used to calculate the new parity data. Id., col. 2 lines 50-60. Note that parity data, which is on a different disk, is updated, not CRC bytes.

Prior art methods had initialized every sector of every drive during a complete initialization, which was very time consuming. *Id.*, col. 2 lines 61-67. Instead, before data is written to a block of a particular stripe, Hashemi checks to see if there is a mismatch in the LBA and/or CRC of that block. *Id.*, col. 8 lines 24-33. If there is a mismatch, then it is assumed that the stripe has not been initialized and then the parity data for that stripe is initialized. *Id.*, col. 8 33-37. If there is not a mismatch, then the parity data is updated with a read-modify-write operation. *Id.*, col. 8 lines 47-62.

Accordingly, it is new parity data that is calculated from the old parity data, and not a new CRC that is calculated from an old CRC. Hashemi does mention a CRC/LBA generation/check logic unit 420, but there is no mention of how the CRC is calculated. *Id.*, col. 5 line 56 to col. 6 line 18.

As to the sections of Hashemi cited by the Office Action, at col. 7 line 45 to col. 8 line 2, Hashemi discusses a check of the mismatch condition discussed above. There is no mention of how a CRC is calculated, let alone a CRC based on a new LBA. Also, at col. 8 lines 47-62, Hashemi describes proceeding with a read-modify-write operation to update parity bytes when no mismatch occurs. Note that the new data is written to an existing block of the stripe; therefore, there is no new LBA in any part of the calculation.

Thus, Hashemi does not teach or suggest any functions in the creation of a CRC, let alone the specific functions recited in claim 1. There would be no teaching of how and/or why to make any modification to the circuit 200 of Frederickson. Accordingly, the references alone or in combination do not teach the above limitation.

For at least the reasons stated above, Applicant submits that claim 1 and its dependent claims 2-7 and 20 are allowable over the cited references.

Applicants submit that independent claims 8 and 15 should be allowable for at least this same rationale. Claims 9-14 depend from claim 8; and claims 16-19 and 21 depend from claim 15 and thus derive patentability at least therefrom.

Objection to Claims 12 and 20-21

Claims 12 and 20-21 were objected to as being dependent on a rejected base claim, but were found to be allowable if rewritten in independent form. In view of the foregoing arguments with regard to claims 1, and thus 8 and 15, Applicant respectfully submits that claims 12 and 20-21 are in condition for allowance without being rewritten in independent form. Withdrawal of the objection is respectfully requested.

CONCLUSION

In view of the foregoing, Applicants believe all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at (415) 576-0200.

Respectfully submitted,

/David B. Raczkowski/

David B. Raczkowski
Reg. No. 52,145

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 415-576-0200
Fax: 415-576-0300
DBR/scz
61192393 v1